MSX

truuks en tips

deel 5

M. Buysse



MSX truuks en tips deel 5 M. Buysse

MSX
Truuks en tips
Deel 5

f 24.90

MSX
truuks en tips
deel 5

M. Buysse

uitgeverij STARK-TEXEL
postbus 302 - 1794 ZG Oosterend tel. 02223 - 661

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

MSX

MSX truuks en tips. – Oosterend : Stark-Texel
Dl. 5 / M. Buysse. – Ill
ISBN 90-6398-745-5
SISO 365.3 UDC 681.3.06
Trefw.: Programmeren (computer) MSX (computer).

1e druk 1986
ISBN 90 6398 745 5

INHOUD

Data maker

Het omzetten van een machinetaalprogramma in dataregels is voor veel machinetaalprogrammeurs een heel vervelende klus. Deze klus die soms noodzakelijk is om een machinetaalprogramma te kunnen doorgeven aan een mede-MSX-er of te kunnen opsturen naar een wedstrijd kan echter zeer eenvoudig met het volgende programma.

Voordat het programma gestart wordt, moet men eerst een aantal dataregels maken. Na de instruktiedata moeten er juist geteld 66 letters of cijfers geplaatst worden. De nummering van de regels moet steeds per 1Ø gebeuren. Het aantal regels dat ingegeven moet worden is afhankelijk van de lengte van het machinetaalprogramma. Dit kan men eenvoudig uitrekenen door het aantal bytes dat het programma lang is te delen door 22. Indien dit niet precies op een even aantal regels uitkomt moet men afronden naar boven.
Nadat de voorgaande instrukties zorgvuldig opgevolgd zijn kan men het programma runnen. Op de vraag "lijnnummer?" geeft men de eerste dataregel die men gereserveerd heeft.
Het invullen van de dataregels zelf kan een tijdje duren (naar gelang de lengte van het programma dat omgezet moet worden). Als het programma klaar is met het invullen van de dataregels moet het volledige Basic-programma weggeschreven worden op tape of disk. Het programma moet wel als een ASCII file weggeschreven worden. Daarna kan het programma terug geladen worden en kunnen de regels 1Ø-19Ø verwijderd worden.
Opgelet: Het programma mag niet onderbroken worden tijdens het maken van de data.

De programmalijst:

```
10 REM ****************************
```

```
20 REM * AUTOMATISCHE DATA-MAKER *
30 REM * ------------------------ *
40 REM **************************
50 REM
60 FORI=&HE000TO&HE01B:READ A$:POKEI,VAL
("&H"+A$):NEXT:DEFUSR=&HE000:CLEAR200,&H
DFFF
70 DATA ED,4B,00,E2,21,01,80,5E,23,56,23
,7E,B9,28,03,EB,18,F5,23,7E,B8,20,F8,23,
22,00,E2,C9
80 INPUT"lijn nummer";LN
90 INPUT"begin adres";BE
100 INPUT"eind  adres";EI
110 FORI=BETOEISTEP21
120 POKE&HE200,(LNMOD256)
130 POKE&HE201,INT(LN/256):X=USR(0)
140 A=PEEK(&HE200)+256*PEEK(&HE201):LN=L
N+10
150 POKEA,&H84:POKEA+1,&H20:A=A+2
160 FORI1=ITOI+20:DA$=HEX$((PEEK(I1))
170 IF LEN(DA$)<2 THEN DA$="0"+DA$
180 POKEA,ASC(DA$):POKEA+1,ASC(RIGHT$(DA
$,1)):POKEA+2,&H2C:A=A+3
190 NEXT:POKEA-1,0:NEXT:END
```

ASCII monitor

Het geheugen bekijken kan op veel verschillende manieren gebeuren. Daarvoor zijn reeds verschillende professionele monitors op de markt. Deze monitors hebben dan onder andere de mogelijkheid het geheugen in ASCII formaat op het scherm te plaatsen. Zo ook in het volgende programma.

In dit volgende programma wordt het scherm gebruikt als een raam dat over het geheugen verplaatst kan worden. Op het scherm ziet men dus steeds een deel van het ROM of RAM geheugen in ASCII formaat verschijnen. Het verplaatsen daarvan doet men zeer eenvoudig met de cursor-toetsen of met de joystick. Door een schuine richting aan te geven met de cursor-toetsen of de joystick wordt het programma beëindigd.

Het programma, zoals het hieronder staat afgedrukt, werkt uitsluitend met de cursor-toetsen. Wil men met een joystick werken, dan moet regel 8Ø als volgt verander worden:

8Ø X=USR (1)

Wil men met een joystick in poort 2 werken dan moet regel 8Ø als volgt worden veranderd:

8Ø X=USR (2)

De programmalijst:

```
10 ' ******************
20 ' *  ASCII MONITOR  *
30 ' *  -------------  *
40 ' ******************
```

```
50 '
60 CLEAR200,&HE0FF:SCREEN1
70 FORI=&HE100TO&HE13F:READ A$:POKEI,VAL
("&H"+A$):NEXT:DEF USR=&HE102
80 X=USR(0)
90 DATA 31,7D,3A,F8,F7,CD,D5,00,2A,00,E1
,FE,01,28,11,FE,03,28,16,FE,05
100 DATA 28,15,FE,07,28,17,FE,00,28,E3,C
9,01,20,00,37,3F,ED,42,18,0A,23
110 DATA 18,07,01,20,00,09,18,01,2B,22,0
0,E1,11,00,18,01,00,03,CD,5C,00,18,C1
```

CAPS-LOCK flitser

Hoewel er in deel 1 reeds een routine verscheen om het caps-lock lampje te doen flikkeren staat hieronder een veel kortere manier die heel eenvoudig in een programma opgenomen kan worden. Bij het omschakelen van het caps-lock lampje wordt er echter niet omgeschakeld van hoofdletters naar kleine letters en omgekeerd.

De FOR-NEXT lus in het programma dient alleen om het omschakelen van het caps-lock lampje te vertragen. Het eigenlijke omschakelen gebeurt met de OUT-instruktie in regel nr. 6∅.

De programmalijst:

```
10 '**********************
20 '*  CAPS LOCK FLITSER  *
30 '*  ----------------   *
40 '**********************
50 '
60 OUT&HAA,INP(&HAA)XOR&H40
70 FORI=1TO100:NEXT:GOTO60
```

Om het caps-lock lampje aan te schakelen kan het volgende programma gebruikt worden.

De programmalijst:

```
10 '******************
20 '*  CAPS LOCK AAN  *
30 '*  ------------   *
40 '******************
```

```
50 '
60 OUT&HAA,INP(&HAA)AND&HBF
```

Het terug uitschakelen van het caps-lock lampje kan met het volgende programma.

De programmalijst:

```
10 '*******************
20 '*  CAPS LOCK UIT  *
30 '*  -------------  *
40 '*******************
50 '
60 OUT&HAA,INP(&HAA)OR&H40
```

String funktie decoder

Verschillende MSX programmeurs onder u zullen reeds eerder geprobeerd hebben een programma te maken dat rekenkundige funkties kan omzetten in een grafiek. Het grote probleem dat daarbij optreedt is dat de betreffende funktie, die met DEF FN gedefinieerd wordt, niet variabel is. Daardoor moet men steeds weer in de listing de funktie aanpassen als men een andere grafiek wil bekijken. Door de volgende subroutine in te bouwen in uw programma is het mogelijk tijdens het lopen van het programma de funktie te veranderen.

De volgende subroutine heeft echter wel nog een klein steuntje nodig. Dit "steuntje" is, dat de regel (waar men normaal de instruktie DEF FN zou plaatsen) het REM-statement bevat met daar achter ongeveer 250 karakters. Als de subroutine aangeroepen wordt moet de variabele A gevuld zijn met het lijnnummer van de betreffende regel.
Na het aanroepen van deze subroutine is het REM-statement veranderd in DEF FNST(X,Y) = (de funktie die ingegeven is). Deze funktie kan daarna steeds in uw programma aangeroepen worden zoals een gewoon gedefinieerde funktie. De naam van de funktie is FNST.

Opgelet: Als de voorgaande richtlijnen niet strikt opgevolgd worden is het zeer waarschijnlijk dat de computer zichzelf op slot zet.

De programmalijst:

```
10 FORI=&HE000TO&HE01B:READ A$:POKEI,VAL
("&H"+A$):NEXT:DEFUSR=&HE000:CLEAR200,&H
DFFF
```

```
20 DATA ED,4B,00,E2,21,01,80,5E,23,56,23
,7E,B9,28,03,EB,18,F5,23,7E,B8,20,F8,23,
22,00,E2,C9
30 REM
40 REM ***************************
50 REM *  STRING-FUNKTIE DECODER  *
60 REM *  ----------------------  *
70 REM ***************************
80 REM
10000 POKE&HE200,(AMOD256)
10010 POKE&HE201,INT(A/256)
10020 X=USR(0)
10030 A=PEEK(&HE200)+256*PEEK(&HE201)
10040 INPUT "FUNKTIE";E$:T1=LEN (E$):T2=
0
10050 POKEA,&H97:POKEA+1,&HDE:POKEA+2,83
:POKEA+3,84:POKEA+4,40:POKEA+5,88:POKEA+
6,44:POKEA+7,89:POKEA+8,41:POKEA+9,&HEF:
A=A+10
10060 IF T1=0 THEN 10250
10070 IF ASC(E$)=43 THEN POKEA,&HF1:A=A+
1:T1=T1-1:E$=RIGHT$(E$,T1):GOTO10060
10080 IF ASC(E$)=45 THEN POKEA,&HF2:A=A+
1:T1=T1-1:E$=RIGHT$(E$,T1):GOTO10060
10090 IF ASC(E$)=42 THEN POKEA,&HF3:A=A+
1:T1=T1-1:E$=RIGHT$(E$,T1):GOTO10060
10100 IF ASC(E$)=47 THEN POKEA,&HF4:A=A+
1:T1=T1-1:E$=RIGHT$(E$,T1):GOTO10060
10110 IF ASC(E$)=94 THEN POKEA,&HF5:A=A+
1:T1=T1-1:E$=RIGHT$(E$,T1):GOTO10060
10120 IF LEN(E$)<3 THEN 10240
10130 EE$=LEFT$(E$,3)
10140 IF EE$="COS" THEN POKEA,&HFF:POKE
A+1,&H8C:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10150 IF EE$="SIN" THEN POKEA,&HFF:POKE
```

```
A+1,&H89:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10160 IF EE$="TAN" THEN POKEA,&HFF:POKE
A+1,&H8D:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10170 IF EE$="SQR" THEN POKEA,&HFF:POKE
A+1,&H87:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10180 IF EE$="ABS" THEN POKEA,&HFF:POKE
A+1,&H86:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10190 IF EE$="ATN" THEN POKEA,&HFF:POKE
A+1,&H8E:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10200 IF EE$="EXP" THEN POKEA,&HFF:POKE
A+1,&H8B:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10210 IF EE$="INT" THEN POKEA,&HFF:POKE
A+1,&H85:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10220 IF EE$="LOG" THEN POKEA,&HFF:POKE
A+1,&H8A:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10230 IF EE$="SGN" THEN POKEA,&HFF:POKE
A+1,&H84:A=A+2:T1=T1-3:E$=RIGHT$(E$,T1):
GOTO 10060
10240 POKEA,ASC(E$):A=A+1:T1=T1-1:E$=RIG
HT$(E$,T1):GOTO10060
10250 POKEA,58:POKEA+1,&H8F
10260 RETURN
```

Karakter editor

Bij het programmeren van spelletjes en andere programma's is het soms gewenst andere karakters te gebruiken. Nu is de karakterset van de MSX-computer in het video-RAM gelegen. Daardoor is het mogelijk deze karakters stuk voor stuk te veranderen (met uitzondering van de cursor).
Om dit herprogrammeren te vergemakkelijken kan onderstaand programma gebruikt worden.

De programmalijst:

```
10 ' *********************
20 ' *  KARAKTER EDITOR  *
30 ' *  ---------------  *
40 ' *********************
50 '
60 CLEAR200,&HD8DA:SCREEN0:COLOR 15,4,7:
WIDTH40
70 PRINT"Geef eerst het te veranderen ka
rakter in";
80 PRINT"door het betreffende karakter i
n te"
90 PRINT"toetsen"
100 INPUTA$
110 IF ASC(A$)=1 OR ASC(A$)>31 THEN 120
ELSE 100
120 A=VDP(4)*&H800+(ASC(A$)*8):IF ASC(A$
)=1 THEN A=(VDP(4)*&H800)+((ASC(MID$(A$,
2,1))-64)*8)
130 CLS:PRINT"Geef nu de bitpatronen bin
air in.":FOR I=65TO95:PRINTCHR$(1);CHR$(
I);:NEXT:FORI=32 TO 255:PRINTCHR$(I);:NE
```

```
XT
140 PRINTAA$:FORI=0TO7:LOCATE0,I+10:PRIN
TI+1;:C$="":FOR T=1 TO 8
150 I$=INKEY$:IF I$="" THEN 150
160 IF ASC(I$)=8 THEN IF T>1 THEN C$=LEF
T$(C$,LEN(C$)-1):T=T-1:PRINTCHR$(8);" ";
CHR$(8);
170 IF ASC(I$)<48 OR ASC(I$)>49 THEN 150
180 PRINTI$;:C$=C$+I$:NEXT
190 VPOKE A+I,VAL("&B"+C$):NEXT
200 CLS:PRINT"nog een karakter verandere
n (j/n)"
210 I$=INKEY$:IF I$="" THEN 210
220 IF I$="j" OR I$="J" THEN CLS:GOTO 70
230 DATA 21,00,08,01,00,08,11,00,D9,CD,5
9,00,C9,21,00,D9,01,00,08,3A,AF,FC,FE,00
,28,05,11,00,00,18,03,11,00,08,CD,5C,00,
C9
240 FORI=&HD8DA TO &HD8FF :READ A$:POKE
I,VAL("&H"+A$):NEXT
250 DEF USR=&HD8DA :DEF USR1=&HD8E7
260 X=USR0(0)
270 ' BSAVE"CAS:K.SET",&HD8E7,&HE100
```

Nadat men klaar is met het maken van een nieuwe karakterset, is het mogelijk deze op tape of disk te saven door middel van:

BSAVE"CAS:naam",&DH8E7,&HE1ØØ

Het probleem dat veelal optreedt met zelfgemaakte karaktersets is dat na het omschakelen van de schermmode de zelfgemaakte karakterset verdwenen is. Bij het gebruik van deze karakter-editor wordt dit probleem echter doeltreffend opgevangen door het gebruik van een klein stukje machinetaal. Nadat men van schermmode is veranderd kan men door dit stukje machinetaal aan te roepen de zelfgemaakte karak-

terset opnieuw in het video-RAM plaatsen. Dit aanroepen gebeurt eenvoudig door in uw programma waar u de zelfgemaakte karakterset wilt gebruiken, de volgende regel op te nemen:

```
1ØCLEAR 2ØØ, &HD8E6: DEF USR1=&HD8E7
```

Waarna men na elke SCREEN instruktie de volgende instruktie plaatst:

```
X=USR (0)
```

De zelfgemaakte karakterset plaatst zich dan automatisch op de goede geheugenplaatsen in het video-RAM. Als men overschakelt naar een grafisch scherm kan de zelfgemaakte karakterset niet gebruikt worden.
Bij het saven van de zelfgemaakte karakterset wordt het stukje machinetaal eveneens opgenomen!

Karakters herzetten

Bij het gebruik van een zelfgemaakte karakterset is het soms wenselijk om terug te kunnen schakelen naar de normale karakterset zonder dat men de "SCREEN" opdracht moet aanroepen. Dit omdat het scherm bijvoorbeeld niet gewist mag worden. Om dat te verhelpen kan men het volgende programma in uw eigen programma inbouwen.

De programmalijst:

```
10 ' ****************************
20 ' *  KARAKTERSET HERZETTEN   *
30 ' *  ---------------------   *
40 ' ****************************
50 '
60 DEF USR=&H71E
70 X=USR(0)
```

Karakters vergroot printen

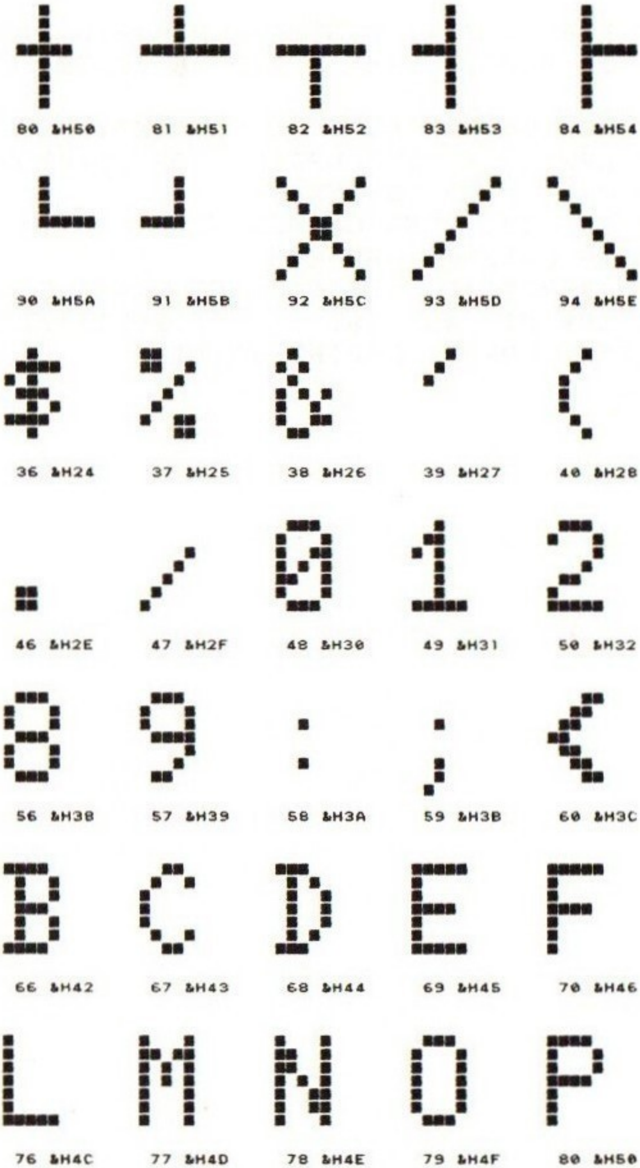
Als men het programma "karakter editor" reeds ingetikt heeft zal men zeker reeds gedacht hebben hoe eenvoudig het zou zijn als men deze karakters ook eens kon uitprinten. Dit uitprinten van zelfgemaakte karakters is niet mogelijk met een eenvoudige printer, omdat bij deze printers de karakterset reeds in het geheugen van de printer zelf is ingeprogrammeerd. Met onderstaand programma is dit echter wel mogelijk.

Al de karakters worden vergroot uitgeprint met onder elk karakter de ASCII waarde in decimaal en in hexadecimaal. De karakterset die uitgeprint wordt is de karakterset die op het moment van het starten van het programma in het video-RAM geplaatst is. Men kan dus evengoed de gewone karakterset uitprinten. Een voorbeeld op pagina 22.

De programmalijst:

```
10 ' ********************************
20 ' *  KARAKTERS VERGROOT PRINTEN  *
30 ' *  --------------------------  *
40 ' ********************************
50 '
60 LPRINTCHR$(27);"Q";
70 LPRINTCHR$(27);"CS";
80 LPRINTCHR$(27);"T10";
90 FOR T1=0TO2047 STEP 80
100 FOR T2=0TO7
110 FOR T3=1TO10
120 A=VPEEK((2048+T1+T2)+(T3*8))
130 IF (T1\8)+T3=>256 THEN 170
```

```
140 FOR T4=7 TO 0 STEP-1:B=A AND 2^T4
150 IF B=2^T4 THEN LPRINT"█";ELSE LPRINT
" ";
160 NEXTT4:LPRINT "    ";:NEXTT3
170 LPRINT:NEXTT2:LPRINT:FOR T5=0TO9
180 C=T1/8+T5:C$=HEX$(C):
190 IF C<31 THEN C=C+65 ELSE C=C+1
200 IF C=>256 THEN END
210 C$=HEX$(C):LPRINTUSING "###";C;
220 LPRINT" &H";C$;"    ";:NEXTT5:LPRINT
:LPRINT:LPRINT:LPRINT:NEXTT1
```


80 &H50
81 &H51
82 &H52
83 &H53
84 &H54
90 &H5A
91 &H5B
92 &H5C
93 &H5D
94 &H5E
36 &H24
37 &H25
38 &H26
39 &H27
40 &H28
46 &H2E
47 &H2F
48 &H30
49 &H31
50 &H32
56 &H38
57 &H39
58 &H3A
59 &H3B
60 &H3C
66 &H42
67 &H43
68 &H44
69 &H45
70 &H46
76 &H4C
77 &H4D
78 &H4E
79 &H4F
80 &H50

Truuks & tips laser show

Hoewel er geen laser in uw MSX-computer is ingebouwd kan men wel enkele laser effekten op het scherm waarnemen na het intikken van onderstaand beeldgrapje.
Het geheel werkt anders dan de grote meerderheid van de beeldgrapjes, in het tekstscherm screen Ø.

De programmalijst:

```
10 ' ******************************
20 ' *  TRUUKS & TIPS LASER SHOW  *
30 ' *  ------------------------  *
40 ' ******************************
50 '
60 KEY1,"TRUUKS":KEY2,"& TIPS":KEY3,"":K
EY4,"LASER":KEY5,"SHOW"
70 CLS:T=T+1:IFT>150THENT=2
80 FORI=0TO920STEPT:VPOKEI,22:NEXT:GOTO7
0
```

Introduktie scherm

Als men een spelletje of een ander programma gemaakt heeft wil men daar graag een mooi origineel introduktiescherm bij hebben.
Hoewel er veel goede mogelijkheden zijn om een introduktiescherm te maken, staat hieronder een mooi voorbeeldje.
De tekst in regel nr. 70 kan naar eigen believen veranderd worden.

De programmalijst:

```
10 ' ***********************
20 ' *  INTRODUCTIE SCHERM  *
30 ' *  -----------------   *
40 ' ***********************
50 '
60 SCREEN0:WIDTH40
70 A$=SPACE$(39)+"Dit is een introduktie
-scherm uit TRUUKS & TIPS"
80 FORI=LEN(A$)TO1STEP-1:PRINTMID$(A$,I,
39)
90 FORT=1TO50:NEXTT,I:GOTO80
```

Funktie-toetsen herzetten

Het hiervolgende programma dient om de funktie-toetsen hun normale waarde terug te geven. Dit kan bijvoorbeeld ingebouwd worden in een programma waar de funktie-toetsen een andere inhoud krijgen. Zodat bij het einde van het programma de funktie-toetsen hun normale inhoud terug krijgen.

De programmalijst:

```
10 ' ******************************
20 ' *  FUNKTIE-TOETSEN HERZETTEN *
30 ' *                           *
40 ' ******************************
50 '
60 DEF USR=&H3E
70 X=USR(0):CLS
```

Toets BEEP

Velen onder u zullen reeds eerder programma's uit boeken of tijdschriften ingetikt hebben. Vooral als het om lange programma's gaat, kan men daar menig uurtje aan besteden. Om dit werk een beetje gemakkelijker te maken kan men het volgende programma gebruiken. Nadat het programma opgestart is geeft de computer bij elke toets-aanslag een beepje. Om de beepjes te stoppen tikt men:

POKE &HFDC2, &HC9

De programmalijst:

```
10 ' ***************
20 ' *  TOETS BEEP  *
30 ' *  ----------  *
40 ' ***************
50 '
60 POKE&HFDC2,&HCD:POKE&HFDC3,&HC0:POKE&
HFDC4,0
```

Verdwenen foutmeldingen

Als beginnende Basic-programmeur zult u waarschijnlijk al die vervelende foutmeldingen wel eens beu zijn.
Er bestaat echter een mogelijkheid om de foutmeldingen te laten verdwijnen. Na het intikken en opstarten van het volgende programma bent u van alle foutmeldingen verlost.
Indien u na een bepaalde tijd toch weer heimwee krijgt naar die foutmeldingen, tikt u:

POKE &HFEFD, &HC9

De programmalijst:

```
10 ' ****************************
20 ' *  VERDWENEN FOUTMELDINGEN  *
30 ' *  ----------------------   *
40 ' ****************************
50 '
60 POKE&HFEFD,&HCD:POKE&HFEFE,&H28:POKE&
HFEFF,&H41
```

Simultane output nr 1

Toen in de jaren zeventig de eerste computers op de markt kwamen konden deze nog niet op een televiesietoestel aangesloten worden. Daardoor kwam alle tekst die men intikte en die de computer verschafte op een printer. Bij de MSX-computer daarentegen komt alle tekst op het scherm te staan. Om nu alle tekst ook nog eens op de printer te krijgen dient men het volgende programma in te tikken en op te starten, waarna alle tekst die op het scherm verschijnt ook uitgeprint zal worden op de printer.
Om het uitprinten te beëindigen tikt men:

POKE &HFDA4, &HC9

De programmalijst:

```
10 '***************************
20 '*  SIMULTANE OUTPUT nr 1  *
30 '*  ---------------------  *
40 '***************************
50 '
60 POKE&HFDA4,&HCD:POKE&HFDA5,&HA5:POKE&
HFDA6,&H0
```

Simultane output nr 2

Soms kan het gebeuren dat de tekst die naar de printer gestuurd wordt ook op het scherm moet verschijnen. Dit kan zeer eenvoudig gebeuren door onderstaand programma in te tikken en op te starten. Men kan het natuurlijk ook in een zelfgemaakt programma inbouwen. Om te zorgen dat de tekst niet meer op het scherm verschijnt als er uitgeprint wordt geeft men:

POKE &HFFB6, &HC9

Waarschuwing: Dit programma mag niet gebruikt worden in kombinatie met het programma Simultane Output nr 1. Doet men dit toch dan zal de computer zichzelf op slot zetten.

De programmalijst:

```
10 '*************************
20 '*  SIMULTANE OUTPUT nr 2  *
30 '*  ---------------------  *
40 '*************************
50 '
60 POKE&HFFB6,&HCD:POKE&HFFB7,&HA2:POKE&
HFFB8,&H0
```

Schrijfmachine

Zoals men al zag men het programma Simultane Output nr 1 is het mogelijk alle tekst die op het scherm verschijnt ook op de printer te laten uitprinten. Nu is het niet altijd de bedoeling dat de foutmeldingen en de prompt ook uitgeprint worden. Daarom het volgende programma dat enkel die tekst op de printer uitprint, die men zelf intikt.
Om het uitprinten terug te stoppen, tikt men:

POKE &HFDC2, &HC9

De programmalijst:

```
10 '********************
20 '*  SCHRIJFMACHINE  *
30 '*  --------------  *
40 '********************
50 '
60 POKE&HFDC2,&HCD:POKE&HFDC3,&HA5:POKE&
HFDC4,&H0
```

Celsius fahrenheit

Hoewel in het grootste deel van de wereld de temperatuur in graden celsius gegeven wordt, gebruikt men toch soms nog graden fahrenheit. Daarom het volgende programma dat graden celsius omrekent in graden fahrenheit.

De programmalijst:

```
10 ' *************************
20 ' *  CELSIUS > FAHRENHEIT  *
30 ' *  --------------------  *
40 ' *************************
50 '
60 INPUT"Graden in celsius";A
70 B=A*9/5+32
80 PRINTA;"graden celsius =";B;"graden f
ahrenheit"
```

Fahrenheit celsius

Hoewel in het grootste deel van de wereld de temperatuur in graden celsius gegeven wordt, gebruikt men toch soms nog graden fahrenheit. Daarom het volgende programma dat graden fahrenheit omrekent in graden celcius.

De programmalijst:

```
10 ' **************************
20 ' *  FAHRENHEIT > CELSIUS  *
30 ' *  --------------------  *
40 ' **************************
50 '
60 INPUT"Graden in fahrenheit";B
70 A=(B-32)*5/9
80 PRINTB;"graden fahrenheit =";A;"grade
n celsius"
```

Nederlandse vlag

Het volgende programma is bedoeld voor de vaderlandlievende MSX-er. Na het opstarten van het programma moet men enkele sekonden geduld oefenen. Na enige tijd stopt het programma, en men bevindt zich in "SCREEN 1" met in de plaats van een gewone eenkleurige achtergrond, een achtergrond in de nederlandse driekleur. Deze driekleur blijft eveneens behouden na een "CLS" opdracht. Na een "SCREEN" opdracht echter is de nederlandse driekleur verdwenen.

De programmalijst:

```
10 ' **********************
20 ' *  NEDERLANDSE VLAG  *
30 ' *  ---------------   *
40 ' **********************
50 '
60 SCREEN1:VDP(0)=2:VDP(1)=167:VDP(3)=25
5:VDP(4)=0
70 FORI=8192TO14335
80 VPOKEI,16
90 NEXTI:VDP(1)=231:FORI=0TO7:VPOKE8448+
I,6
100 VPOKE10496+I,15:VPOKE12544+I,4:NEXT
```

Elastisch scherm

Het scherm verwijden en versmallen gaat zeer eenvoudig met de instruktie "WIDTH". Het scherm kan echter ook verkort en verlengd worden.
Als het scherm verkort wordt kan men dit op het beeld zien. Als het scherm echter verlengd wordt en daardoor langer dan 24 regels wordt, ziet men dit niet meer. De ruimte die dan zogezegd onder het beeld gevallen is kan wel gevuld worden met tekst maar is onzichtbaar. Ook de instruktie "LOCATE" kan onder het beeld gebruikt worden.
Met het volgende programma kan het verkorten en verlengen van het scherm zeer eenvoudig gedaan worden.

De programmalijst:

```
10 ' **********************
20 ' *  ELASTISCH SCHERM  *
30 ' *  ----------------  *
40 ' **********************
50 '
60 INPUT"BREEDTE (max 255)";A
70 POKE 62385!,A
```

Tekst op zijn kop

Het volgende programma keert alle tekst op zijn kop. Dit gebeurt zowel als men in "SCREEN0" staat als in "SCREEN 1". Om terug over te schakelen naar de gewone karakters, moet men het programma nogmaals opstarten.

De programmalijst:

```
10 ' ***********************
20 ' *  TEKST OP ZIJN KOP  *
30 ' *  ----------------   *
40 ' ***********************
50 '
60 T=VDP(4)*2048
70 FORI=T TO T+2040 STEP 8
80 FORJ=0TO7:A(J)=VPEEK(I+J):NEXTJ
90 FORJ=0TO7:VPOKEI+J,A(7-J):NEXTJ,I
```

SCREEN 0
spiraalsgewijs wissen

Het scherm wissen kan zeer eenvoudig met de instruktie "CLS". Dit gebeurt dan op een zo snelle wijze dat men er niets van ziet. Om het schermwissen op een iets mooiere wijze te laten verlopen kan men het volgende programma gebruiken.

Dit programma wist "SCREEN ∅" spiraalsgewijs. Men kan het programma desgewenst inbouwen in een zelfgemaakt programma.

De programmalijst:

```
10 ' ***************************
20 ' *  SPIRAALSGEWIJS WISSEN  *
30 ' *      VAN SCREEN 0       *
40 ' *  ---------------------  *
50 ' ***************************
60 '
70 FORI=0TO11
80 FORT=ITO39-I:VPOKE(I*40)+T,32:NEXT
90 FORT=39*ITO((39-I)*(23-I))+39*ISTEP 4
0:VPOKET,32:NEXT
100 FORT=39-ITOISTEP-1:VPOKE920-(I*40)+T
,32:NEXT
110 FORT=920+ITOI*40STEP-40:VPOKET,32:NE
XT
120 NEXT
```

SCREEN 1
spiraalsgewijs wissen

Het scherm wissen kan zeer eenvoudig met de instruktie "CLS". Dit gebeurt dan op een zo snelle wijze dat men er niets van ziet. Om het schermwissen op een iets mooiere wijze te laten verlopen kan men het volgende programma gebruiken.

Dit programma wist "SCREEN 1" spiraalsgewijs. Men kan het programma desgewenst inbouwen in een zelfgemaakt programma.

De programmalijst:

```
10 ' ***************************
20 ' *  SPIRAALSGEWIJS WISSEN  *
30 ' *      VAN SCREEN 1       *
40 ' *  ---------------------  *
50 ' ***************************
60 '
70 FORI=0TO11
80 FORT=ITO31-I:VPOKE(I*32)+T+6144,32:NE
XT
90 FORT=31*ITO((31-I)*(23-I))+31*ISTEP 3
2:VPOKET+6144,32:NEXT
100 FORT=31-ITOISTEP-1:VPOKE736-(I*32)+T
+6144,32:NEXT
110 FORT=736+ITOI*32STEP-32:VPOKET+6144,
32:NEXT
120 NEXT
```

SCREEN 2 spiraalsgewijs wissen

Het scherm wissen kan zeer eenvoudig met de instruktie "CLS". Dit gebeurt dan op een zo snelle wijze dat men er niets van ziet. Om het schermwissen op een iets mooiere wijze te laten verlopen kan men het volgende programma gebruiken.

Dit programma wist "SCREEN 2" spiraalsgewijs. Men kan het programma desgewenst inbouwen in een zelfgemaakt programma.

De programmalijst:

```
10 ' ****************************
20 ' *  SPIRAALSGEWIJS WISSEN   *
30 ' *      VAN SCREEN 2        *
40 ' *  ---------------------   *
50 ' ****************************
60 '
70 FORI=0TO95
80 LINE(I,I)-(255-I,191-I),PEEK(&HF3EA),
B
90 NEXT
```

SCREEN 3
spiraalsgewijs wissen

Het scherm wissen kan zeer eenvoudig met de instruktie "CLS". Dit gebeurt dan op een zo snelle wijze dat men er niets van ziet. Om het schermwissen op een iets mooiere wijze te laten verlopen kan men het volgende programma gebruiken.

Dit programma wist "SCREEN 3" spiraalsgewijs. Men kan het programma desgewenst inbouwen in een zelfgemaakt programma.

De programmalijst:

```
10 ' ***************************
20 ' *  SPIRAALSGEWIJS WISSEN  *
30 ' *      VAN SCREEN 3       *
40 ' *  ---------------------  *
50 ' ***************************
60 '
70 FORI=0TO92STEP4
80 LINE(I,I)-(255-I,191-I),PEEK(&HF3EA),
B
90 NEXT
```

Cosecans

Hoewel er in de MSX-basic reeds veel wiskundige funkties ingebouwd zijn kan het toch soms nog gebeuren dat men een bepaalde wiskundige funktie meerdere malen nodig heeft in een programma. Deze wiskundige funktie kan men dan zelf definiëren met de instruktie "DEF FN". Maar, en dat is zeer belangrijk, men moet de juiste formule hebben.

Het onderstaande programma toont hoe men de wiskundige funktie "COSECANS" kan definiëren in een programma.

De programmalijst:

```
10 ' **************
20 ' *  COSECANS  *
30 ' *  --------  *
40 ' **************
50 '
60 DEF FNTT(X)= 1/SIN(X)
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

SECANS

Het onderstaande programma toont hoe men de wiskundige funktie "SECANS" kan definiëren in een programma.

De programmalijst:

```
10 ' ************
20 ' *  SECANS  *
```

```
30 ' *  ------  *
40 ' ************
50 '
60 DEF FNTT(X)= 1/COS(X)
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

COTANGENS

Het onderstaande programma toont hoe men de wiskundige funktie "COTANGENS" kan definiëren in een programma.

De programmalijst:

```
10 ' ***************
20 ' *  COTANGENS  *
30 ' *  ---------  *
40 ' ***************
50 '
60 DEF FNTT(X)= 1/TAN(X)
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

Inverse cosecans

Hoewel er in de MSX-basic reeds veel wiskundige funkties ingebouwd zijn kan het toch soms nog gebeuren dat men een bepaalde wiskundige funktie meerdere malen nodig heeft in een programma. Deze wiskundige funktie kan men dan zelf definiëren met de instruktie "DEF FN". Maar, en dat is zeer belangrijk, men moet de juiste formule hebben.

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COSECANS" kan definiëren in een programma.

De programmalijst:

```
10 ' **********************
20 ' *  INVERSE COSECANS  *
30 ' *  ----------------  *
40 ' **********************
50 '
60 DEF FNTT(X)= ((SGN(X)-1)*1.5708)+ATN(
X/SQR(X*X-1))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE SECANS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE SECANS" kan definiëren in een programma.

De programmalijst:

```
10 ' ********************
20 ' *  INVERSE SECANS   *
30 ' *  -------------    *
40 ' ********************
50 '
60 DEF FNTT(X)= (SGN(SGN(X)-1)*1.5708)+A
TN(X/SQR(X*X-1))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE COTANGENS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COTANGENS" kan definiëren in een programma.

De programmalijst:

```
10 ' ***********************
20 ' *  INVERSE COTANGENS   *
30 ' *  ----------------    *
40 ' ***********************
50 '
60 DEF FNTT(X)= ATN(X)+1.5708
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE COSINUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COSINUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ********************
20 ' *  INVERSE COSINUS  *
30 ' *  ---------------  *
40 ' ********************
50 '
60 DEF FNTT(X)= -ATN(X/SQR(-X*X+1))+1.57
08
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE SINUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE SINUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ******************
20 ' *  INVERSE SINUS  *
30 ' *  -------------  *
40 ' ******************
50 '
60 DEF FNTT(X)= ATN(X/SQR(-X*X+1))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

Sinus hyperbolicus

Hoewel er in de MSX-basic reeds veel wiskundige funkties ingebouwd zijn kan het toch soms nog gebeuren dat men een bepaalde wiskundige funktie meerdere malen nodig heeft in een programma. Deze wiskundige funktie kan men dan zelf definiëren met de instruktie "DEF FN". Maar, en dat is zeer belangrijk, men moet de juiste formule hebben.

Het onderstaande programma toont hoe men de wiskundige funktie "SINUS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ************************
20 ' *  SINUS HYPERBOLICUS  *
30 ' *  ------------------  *
40 ' ************************
50 '
60 DEF FNTT(X)= (EXP(X)-EXP(-X))/2
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

COSINUS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "COSINUS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ************************
20 ' *  COSINUS HYPERBOLICUS  *
30 ' *  ------------------   *
40 ' ************************
50 '
60 DEF FNTT(X)= (EXP(X)+EXP(-X))/2
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

SECANS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "SECANS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ***********************
20 ' *  SECANS HYPERBOLICUS  *
30 ' *  -----------------   *
40 ' ***********************
50 '
60 DEF FNTT(X)= 2/(EXP(X)+EXP(-X))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

COSECANS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "COSECANS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ***************************
20 ' *  COSECANS HYPERBOLICUS  *
30 ' *  ---------------------  *
40 ' ***************************
50 '
60 DEF FNTT(X)= 2/(EXP(X)-EXP(-X))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

TANGENS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "TANGENS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' **************************
20 ' *  TANGENS HYPERBOLICUS  *
30 ' *  --------------------  *
40 ' **************************
50 '
60 DEF FNTT(X)= EXP(-X)/(EXP(X)+EXP(-X))
*2+1
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

COTANGENS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "COTANGENS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 ' ***************************
20 ' *  COTANGENS HYPERBOLICUS  *
30 ' *  ----------------------  *
40 ' ***************************
50 '
60 DEF FNTT(X)= EXP(-X)/(EXP(X)-EXP(-X))
*2+1
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

Inverse tangens hyperbolicus

Hoewel er in de MSX-basic reeds veel wiskundige funkties ingebouwd zijn kan het toch soms nog gebeuren dat men een bepaalde wiskundige funktie meerdere malen nodig heeft in een programma. Deze wiskundige funktie kan men dan zelf definiëren met de instruktie "DEF FN". Maar, en dat is zeer belangrijk, men moet de juiste formule hebben.

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE TANGENS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 '******************************
20 '* INVERSE TANGENS HYPERBOLICUS *
30 '* ---------------------------- *
40 '******************************
50 '
60 DEF FNTT(X)= LOG((1+X)/(1-X))/2
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE COTANGENS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COTANGENS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 '********************************
20 '*INVERSE COTANGENS HYPERBOLICUS*
30 '*------------------------------*
40 '********************************
50 '
60 DEF FNTT(X)= LOG((1+X)/(X-1))/2
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE SINUS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE SINUS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 '********************************
20 '*  INVERSE SINUS HYPERBOLICUS  *
30 '*  --------------------------  *
40 '********************************
50 '
60 DEF FNTT(X)= LOG(X+SQR(X*X+1))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE COSINUS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COSINUS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 '*******************************
20 '* INVERSE COSINUS HYPERBOLICUS *
30 '* --------------------------- *
40 '*******************************
50 '
60 DEF FNTT(X)= LOG(X+SQR(X*X-1))
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE SECANS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE SECANS HYPERBOLICUS" kan definiëren in een programma.

De programmalijst:

```
10 '******************************
20 '* INVERSE SECANS HYPERBOLICUS *
30 '* -------------------------- *
40 '******************************
50 '
60 DEF FNTT(X)= LOG((SQR(-X*X+1)+1)/X)
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

INVERSE COSECANS HYPERBOLICUS

Het onderstaande programma toont hoe men de wiskundige funktie "INVERSE COSECANS HYPERBOLICUS " kan definiëren in een programma.

De programmalijst:

```
10 '******************************
20 '*INVERSE COSECANS HYPERBOLICUS*
30 '*----------------------------*
40 '******************************
50 '
60 DEF FNTT(X)= LOG((SGN(X)*SQR(X*X+1)+1
)/X)
70 INPUT A
80 PRINT FNTT(A)
90 GOTO70
```

Stop het printen

Het volgende programma is bedoeld voor de MSX-er die nog geen printer heeft. Regelmatig ziet men namelijk programma's die gebruik maken van een printer. Als men nu geen printer heeft en het printen is toch gestart in het programma dan zet de computer zich over het algemeen op slot. Om te zorgen dat de computer zich niet op slot zet kan men eerst het volgende programma opstarten.

De programmalijst:

```
10 ' **********************
20 ' *  STOP HET PRINTEN  *
30 ' *  ---------------   *
40 ' **********************
50 '
60 POKE&HFFB6,&H33:POKE&HFFB7,&H33:POKE&
HFFB8,C9
70 '
80 ' *************************
90 ' *  Opnieuw printen met:  *
100 '*                       *
110 '*   POKE&HFFB6,&HC9     *
120 '*************************
```

Verboden te runnen

Het volgende programma schakelt het "RUN"-kommando uit. Dit kan vooral handig gebruikt worden als men een lange listing aan het intikken is. Als men dan toch per toeval op F5 (RUN) drukt wordt het programma niet opgestart en gaat het programma niet verloren. Vooral bij machinetaalprogramma's is dit handig.

De programmalijst:

```
10 ' ************************
20 ' *  VERBODEN TE RUNNEN  *
30 ' *  ------------------  *
40 ' ************************
50 '
60 POKE&HFECB,&H33:POKE&HFECC,&H33:POKE&
HFECD,C9
70 '
80 ' *************************
90 ' *  Opnieuw runnen met:  *
100 '*                       *
110 '*   POKE&HFECB,&HC9     *
120 '*************************
```

Programmabeveiliging

Bij de aankoop van een commercieel programma merkt men steeds dat deze programma's beveiligd zijn. Soms kan het gebeuren dat men een zelfgemaakt programma evengoed wil beveiligen. Daarvoor moet men sommige truuks kennen. In het tweede, derde en vierde deeltje werden al enkele van deze truuks besproken. Als men het onderstaande programma inbouwt in een zelfgemaakt programma heeft men weer een sterke beveiliging méér in het programma. Nadat het onderstaande programma in uw programma is ingebouwd moet er in datzelfde programma geen "ON STOP GOSUB" en geen "ON ERROR GOSUB" meer ingebouwd worden. Dit programma zorgt namelijk dat als het programma gestopt wordt, om welke reden dan ook, dat de computer zichzelf op slot zet.
Opgelet: save eerst het programma vóór het opstarten.

De programmalijst:

```
10 ' ***************************
20 ' *  PROGRAMMA BEVEILIGING  *
30 ' *  ---------------------  *
40 ' ***************************
50 '
60 POKE&HFF07,&HCD:POKE&HFF08,&H28:POKE&
HFF09,&H41
```

Verdwenen listing

Na het opstarten van het volgende programma is het "LIST" kommando uitgeschakeld. Dit kan men in een programma inbouwen zodat na het uitvoeren van het programma de gebruiker geen list meer kan opvragen.

De programmalijst:

```
10 ' ***********************
20 ' *  VERDWENEN LISTING  *
30 ' *  ----------------   *
40 ' ***********************
50 '
60 POKE&HFF89,&H33:POKE&HFF8A,&H33:POKE&
HFF8B,C9
70 '
80 ' *************************
90 ' *  Opnieuw listen met:  *
100 '*                       *
110 '*   POKE&HFF89,&HC9     *
120 '*************************
```

Muzikale joystick

Het volgende programma maakt van uw joystick een muzikaal instrument. Elke richting geeft een bepaalde noot en met de vuurknoppen kan men de oktaven verhogen en verlagen.

De programmalijst:

```
10 ' ***********************
20 ' *  MUZIKALE JOYSTICK  *
30 ' *  -----------------  *
40 ' ***********************
50 '
60 ' ******************
70 ' *                *
80 ' * UP          = C *
90 ' * UP+RIGHT    = D *
100 '* RIGHT       = E *
110 '* DOWN+RIGHT  = F *
120 '* DOWN        = G *
130 '* DOWN+LEFT   = A *
140 '* LEFT        = B *
150 '* FIRE 1 = O+     *
160 '* FIRE 2 = O-     *
170 '*                 *
180 '******************
190 '
200 COLOR 15,1:SCREEN0
210 PRINT"***********************"
220 PRINT"*  MUZIKALE JOYSTICK  *"
230 PRINT"*  -----------------  *"
```

```
240 PRINT"************************"
250 O1=4:ON STRIG GOSUB,360,,380
260 STRIG(1) ON:STRIG(3) ON
270 ON STICK(1) GOTO 290,300,310,320,330
,340,350,280
280 IF PLAY(0)=0THEN270 ELSE280
290 PLAY"C":GOTO280
300 PLAY"D":GOTO280
310 PLAY"E":GOTO280
320 PLAY"F":GOTO280
330 PLAY"G":GOTO280
340 PLAY"A":GOTO280
350 PLAY"B":GOTO280
360 IF O1<8 THEN O1=O1+1:PLAY"O=O1;"
370 RETURN
380 IF O1>1 THEN O1=O1-1:PLAY"O=O1;"
390 RETURN
```

Snelle sprites

Vele MSX-ers zullen wel eerder geprobeerd hebben een snel en flitsend aktiespelletje te maken. Als het spelletje dan uiteindelijk klaar is krijgt men over het algemeen een bedroevend resultaat. Hoewel de layout van het spellètje in vele gevallen zeer mooi is zal de snelheid van het spelletje veel te laag zijn. Om toch nog een redelijk snel spel in Basic te kunnen schrijven moet men al veel ervarig hebben en vooral moet men bepaalde truuks kennen. Zo kan men bijvoorbeeld bij al de FOR-NEXT lussen de variabele niet vermelden achter de instruktie "NEXT". Dit geeft al een veel beter resultaat. Wat men ook kan doen is de sprites niet met de instruktie "PUT SPRITE" besturen zoals algemeen gedaan wordt, maar de sprites besturen door rechtstreeks in het video RAM te poken.
In het video RAM is er namelijk een "Sprite attribute table". Dit is het stuk geheugen dat alle gegevens van de sprites bijhoudt. Daar er slechts 32 sprites tegelijkertijd op het scherm kunnen staan is er in dit stuk geheugen slechts plaats voor 32 sprites. Elke sprite heeft 4 bytes die de volgende waarden bevatten:

Byte nr 1 = Y-positie van de sprite
Byte nr 2 = X-positie van de sprite
Byte nr 3 = Het nummer van de sprite
Byte nr 4 = De kleur van de sprite

Het beginadres van de "Sprite attribute table" ligt op geheugenplaats 6912, zodat de vier bytes voor de eerste sprite van geheugenplaats 6912 tot en met geheugenplaats 6915 liggen; de vier volgende bytes zijn voor de tweede sprite, enz.

Het volgende programma geeft duidelijk het verschil van snelheid weer tussen sprites besturen met de instruktie

"PUT SPRITE" en sprites besturen door rechtstreeks in het video RAM te poken.

De programmalijst.

```
10 ' *********************
20 ' *  SNELLE SPRITE'S  *
30 ' * ---------------   *
40 ' *********************
50 '
60 COLOR 15,1,1:SCREEN2
70 FORI=1TO8:READ A
80 S$=S$+CHR$(A)
90 NEXT:SPRITE$(0)=S$
100 DATA 16,56,56,16,120,16,40,68
110 PUT SPRITE0,(0,50),11,0
120 TIME=0:FORI=1TO250
130 PUT SPRITE0,(I,50),11,0
140 NEXT:T1=TIME
150 PUT SPRITE0,(0,50),11,0
160 TIME=0:FORI=1TO250
170 VPOKE6913,I
180 NEXT:T2=TIME
190 SCREEN0
200 PRINT"tijd met 'PUT SPRITE'=";T1
210 PRINT"tijd met 'VPOKE'     =";T2
```

Scroll-routines

In vele van de spelletjes die reeds op de markt zijn ziet men dat het beeld verschuift. Dit verschuiven (scrollen) is echter zeer moeilijk te programmeren. Als men zulke scroll-routines in Basic maakt, loopt het programma veel te traag.
Daarom moeten deze scroll-routines in machinetaal gemaakt worden. Door deze stukjes machinetaal in uw Basic listing op te nemen is het toch mogelijk voor een Basic-programmeur om het beeld te laten scrollen.
In de volgende programma's wordt getoond hoe men zulke scroll-routines in een programma kan inbouwen.

```
10 ' *******************
20 ' *   SCROLL LEFT   *
30 ' *   -----------   *
40 ' *                 *
50 ' * scrolt screen 2 *
60 ' *  naar links     *
70 ' *                 *
80 ' *******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR0=&HE200
120 FORI=&HE200 TO &HE230: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 COLOR 15,1,1:SCREEN 2
140 LINE (50,10)-(200,50),13,B
150 LINE (50,70)-(200,110),13,B
160 LINE (50,130)-(200,170),13,B
170 OPEN"GRP:" AS #1
180 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
190 PRESET(95,90):PRINT#1,"SCROLL"
200 PRESET(88,150):PRINT#1,"ROUTINES"
210 CLOSE#1
220 X=USR0(0):GOTO220
230 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
240 DATA C5,1A,ED,B0,12,13,23,C1,3E,D8,B
C,20,F3,F3,21,00,D5,11,00,18,01
250 DATA 00,03,CD,5C,00,FB,C9
```

```
10 ' ******************
20 ' *  SCROLL LEFT   *
30 ' *  -----------   *
40 ' *                *
50 ' * scrolt screen1 *
60 ' *   naar links   *
70 ' *                *
80 ' ******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR0=&HE200
120 FORI=&HE200 TO &HE230: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
140 A$="&&&&&&&&&&&&&&&&&&&&"
150 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
160 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
170 LOCATE4,11:PRINT"&       SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
180 LOCATE4,19:PRINT"&      ROUTINES     &
":LOCATE4,21:PRINTA$
190 X=USR0(0):GOTO190
200 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
210 DATA C5,1A,ED,B0,12,13,23,C1,3E,D8,B
C,20,F3,F3,21,00,D5,11,00,18,01
220 DATA 00,03,CD,5C,00,FB,C9
```

```
10 ' ******************
20 ' *  SCROLL RIGHT   *
30 ' *  ------------   *
40 ' *                 *
50 ' * scrolt screen 2 *
60 ' *   naar rechts   *
70 ' *                 *
80 ' ******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR0=&HE1A0
120 FORI=&HE1A0 TO &HE1D0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 COLOR 15,1,1:SCREEN 2
140 LINE (50,10)-(200,50),13,B
150 LINE (50,70)-(200,110),13,B
160 LINE (50,130)-(200,170),13,B
170 OPEN"GRP:" AS #1
180 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
190 PRESET(95,90):PRINT#1,"SCROLL"
200 PRESET(88,150):PRINT#1,"ROUTINES"
210 CLOSE#1
220 X=USR0(0):GOTO220
230 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
240 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D5,11,00,18,01
250 DATA 00,03,CD,5C,00,FB,C9
```

```
10 ' ******************
20 ' *  SCROLL RIGHT   *
30 ' *  ------------   *
40 ' *                 *
50 ' * scrolt screen 1 *
60 ' *   naar rechts   *
70 ' *                 *
80 ' ******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR0=&HE1A0
120 FORI=&HE1A0 TO &HE1D0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
140 A$="&&&&&&&&&&&&&&&&&&"
150 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
160 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
170 LOCATE4,11:PRINT"&        SCROLL        &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
180 LOCATE4,19:PRINT"&       ROUTINES       &
":LOCATE4,21:PRINTA$
190 X=USR0(0):GOTO190
200 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
210 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D5,11,00,18,01
220 DATA 00,03,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL TOP RIGHT    *
30 ' *  ----------------    *
40 ' *                      *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 2  *
70 ' *     naar rechts      *
80 ' *                      *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HE150
130 FORI=&HE150 TO &HE180: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR0(0):GOTO230
240 DATA 21,00,18,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
250 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,18,01
260 DATA 00,01,CD,5C,00,FB,C9.
```

```
10 ' **********************
20 ' *  SCROLL TOP RIGHT   *
30 ' *  ----------------   *
40 ' *                    *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 1  *
70 ' *     naar rechts     *
80 ' *                    *
90 ' **********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HE150
130 FORI=&HE150 TO &HE180: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&        SCROLL       &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&       ROUTINES      &
":LOCATE4,21:PRINTA$
200 X=USR0(0):GOTO200
210 DATA 21,00,18,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
220 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,18,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' *************************
20 ' *  SCROLL MIDDLE RIGHT  *
30 ' *  -------------------  *
40 ' *                       *
50 ' * scrolt het middelste  *
60 ' *  deel van screen 2    *
70 ' *     naar rechts       *
80 ' *                       *
90 ' *************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HE100
130 FORI=&HE100 TO &HE130: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR1(0):GOTO230
240 DATA 21,00,19,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
250 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,19,01
260 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL MIDDLE RIGHT  *
30 ' *  -------------------  *
40 ' *                       *
50 ' * scrolt het middelste  *
60 ' *  deel van screen 1 ·  *
70 ' *     naar rechts       *
80 ' *                       *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HE100
130 FORI=&HE100 TO &HE130: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&      ROUTINES    &
":LOCATE4,21:PRINTA$
200 X=USR1(0):GOTO200
210 DATA 21,00,19,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
220 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,19,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ************************
20 ' *  SCROLL BOTTOM RIGHT  *
30 ' *  ------------------   *
40 ' *                      *
50 ' * scrolt het onderste  *
60 ' *  deel van screen 2   *
70 ' *    naar rechts       *
80 ' *                      *
90 ' ************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HE0A0
130 FORI=&HE0A0 TO &HE0D0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR2(0):GOTO230
240 DATA 21,00,1A,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
250 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,1A,01
260 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ************************
20 ' *  SCROLL BOTTOM RIGHT  *
30 ' *  -----------------    *
40 ' *                      *
50 ' * scrolt het onderste   *
60 ' *  deel van screen 1    *
70 ' *     naar rechts       *
80 ' *                      *
90 ' ************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HE0A0
130 FORI=&HE0A0 TO &HE0D0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&       SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&      ROUTINES     &
":LOCATE4,21:PRINTA$
200 X=USR2(0):GOTO200
210 DATA 21,00,1A,11,00,D7,01,00,01,CD,5
9,00,21,FE,D7,11,FF,D7,01,1F,00
220 DATA C5,1A,ED,B8,12,1B,2B,C1,3E,D4,B
C,20,F3,F3,21,00,D7,11,00,1A,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *    SCROLL TOP LEFT    *
30 ' *    ---------------    *
40 ' *                       *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 2   *
70 ' *      naar links       *
80 ' *                       *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HE050
130 FORI=&HE050 TO &HE080: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR0(0):GOTO230
240 DATA 21,00,18,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
250 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,18,01
260 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *   SCROLL TOP LEFT    *
30 ' *   ---------------    *
40 ' *                      *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 1   *
70 ' *      naar links      *
80 ' *                      *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HE050
130 FORI=&HE050 TO &HE080: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&       SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&      ROUTINES     &
":LOCATE4,21:PRINTA$
200 X=USR0(0):GOTO200
210 DATA 21,00,18,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
220 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,18,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL MIDDLE LEFT  *
30 ' *  ------------------  *
40 ' *                      *
50 ' * scrolt het middelste *
60 ' *  deel van screen 2   *
70 ' *     naar links       *
80 ' *                      *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HE000
130 FORI=&HE000 TO &HE030: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR1(0):GOTO230
240 DATA 21,00,19,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
250 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,19,01
260 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ************************
20 ' *  SCROLL MIDDLE LEFT  *
30 ' *  ------------------  *
40 ' *                      *
50 ' * scrolt het middelste *
60 ' *  deel van screen 1   *
70 ' *     naar links       *
80 ' *                      *
90 ' ************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HE000
130 FORI=&HE000 TO &HE030: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL     &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES    &
":LOCATE4,21:PRINTA$
200 X=USR1(0):GOTO200
210 DATA 21,00,19,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
220 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,19,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL BOTTOM LEFT  *
30 ' *  ------------------  *
40 ' *                     *
50 ' * scrolt het onderste  *
60 ' *  deel van screen 2   *
70 ' *     naar links       *
80 ' *                     *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDFA0
130 FORI=&HDFA0 TO &HDFD0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR2(0):GOTO230
240 DATA 21,00,1A,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
250 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,1A,01
260 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL BOTTOM LEFT  *
30 ' *  ------------------  *
40 ' *                     *
50 ' * scrolt het onderste  *
60 ' *  deel van screen 1   *
70 ' *     naar links       *
80 ' *                     *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDFA0
130 FORI=&HDFA0 TO &HDFD0: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES     &
":LOCATE4,21:PRINTA$
200 X=USR2(0):GOTO200
210 DATA 21,00,1A,11,00,D5,01,00,01,CD,5
9,00,21,01,D5,11,00,D5,01,1F,00
220 DATA C5,1A,ED,B0,12,13,23,C1,3E,D6,B
C,20,F3,F3,21,00,D5,11,00,1A,01
230 DATA 00,01,CD,5C,00,FB,C9
```

```
10 ' *********************
20 ' *   SCROLL TOP DOWN   *
30 ' *   ---------------   *
40 ' *                     *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 2  *
70 ' *     naar onder      *
80 ' *                     *
90 ' *********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HDE50
130 FORI=&HDE50 TO &HDE75: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR0(0):GOTO230
240 DATA 21,00,18,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
250 DATA ED,B0,F3,21,00,D5,11,00,18,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *   SCROLL TOP DOWN   *
30 ' *   -------------     *
40 ' *                     *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 1  *
70 ' *     naar onder      *
80 ' *                     *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HDE50
130 FORI=&HDE50 TO &HDE75: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&        SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&       ROUTINES     &
":LOCATE4,21:PRINTA$
200 X=USR0(0):GOTO200
210 DATA 21,00,18,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
220 DATA ED,B0,F3,21,00,D5,11,00,18,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL MIDDLE DOWN  *
30 ' *  ------------------  *
40 ' *                     *
50 ' * scrolt het middenste *
60 ' *  deel van screen 2   *
70 ' *     naar onder      *
80 ' *                     *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HDE00
130 FORI=&HDE00 TO &HDE25: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR1(0):GOTO230
240 DATA 21,00,19,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
250 DATA ED,B0,F3,21,00,D5,11,00,19,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ************************
20 ' *  SCROLL MIDDLE DOWN  *
30 ' *  ------------------  *
40 ' *                      *
50 ' * scrolt het middenste *
60 ' *  deel van screen 1   *
70 ' *     naar onder       *
80 ' *                      *
90 ' ************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HDE00
130 FORI=&HDE00 TO &HDE25: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL     &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES    &
":LOCATE4,21:PRINTA$
200 X=USR1(0):GOTO200
210 DATA 21,00,19,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
220 DATA ED,B0,F3,21,00,D5,11,00,19,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ************************
20 ' *  SCROLL BOTTOM DOWN   *
30 ' *  ------------------   *
40 ' *                      *
50 ' * scrolt het onderste  *
60 ' *  deel van screen 2   *
70 ' *     naar onder       *
80 ' *                      *
90 ' ************************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDDA0
130 FORI=&HDDA0 TO &HDDC5: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR2(0):GOTO230
240 DATA 21,00,1A,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
250 DATA ED,B0,F3,21,00,D5,11,00,1A,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL BOTTOM DOWN  *
30 ' *  ------------------  *
40 ' *                     *
50 ' * scrolt het onderste *
60 ' *  deel van screen 1  *
70 ' *     naar onder      *
80 ' *                     *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDDA0
130 FORI=&HDDA0 TO &HDDC5: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL     &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES    &
":LOCATE4,21:PRINTA$
200 X=USR2(0):GOTO200
210 DATA 21,00,1A,11,20,D5,01,00,01,CD,5
9,00,21,00,D6,11,00,D5,01,1F,00
220 DATA ED,B0,F3,21,00,D5,11,00,1A,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *    SCROLL TOP UP     *
30 ' *    -------------     *
40 ' *                      *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 2  *
70 ' *      naar boven     *
80 ' *                      *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HDF50
130 FORI=&HDF50 TO &HDF75: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR0(0):GOTO230
240 DATA 21,00,18,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
250 DATA ED,B0,F3,21,20,D5,11,00,18,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *     SCROLL TOP UP     *
30 ' *     -------------     *
40 ' *                       *
50 ' * scrolt het bovenste *
60 ' *  deel van screen 1  *
70 ' *     naar boven      *
80 ' *                       *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR0=&HDF50
130 FORI=&HDF50 TO &HDF75: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL     &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES    &
":LOCATE4,21:PRINTA$
200 X=USR0(0):GOTO200
210 DATA 21,00,18,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
220 DATA ED,B0,F3,21,20,D5,11,00,18,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *   SCROLL MIDDLE UP    *
30 ' *   ----------------    *
40 ' *                       *
50 ' * scrolt het middenste *
60 ' *  deel van screen 2    *
70 ' *     naar boven        *
80 ' *                       *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR1=&HDF00
130 FORI=&HDF00 TO &HDF25: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR1(0):GOTO230
240 DATA 21,00,19,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
250 DATA ED,B0,F3,21,20,D5,11,00,19,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *   SCROLL MIDDLE UP   *
30 ' *   ----------------   *
40 ' *                      *
50 ' * scrolt het middenste *
60 ' *  deel van screen 1   *
70 ' *     naar boven       *
80 ' ***********************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR1=&HDF00
120 FORI=&HDF00 TO &HDF25: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
140 A$="&&&&&&&&&&&&&&&&&&&&&&"
150 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
160 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
170 LOCATE4,11:PRINT"&       SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
180 LOCATE4,19:PRINT"&      ROUTINES     &
":LOCATE4,21:PRINTA$
190 X=USR1(0):GOTO190
200 DATA 21,00,19,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
210 DATA ED,B0,F3,21,20,D5,11,00,19,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' **********************
20 ' *  SCROLL BOTTOM UP   *
30 ' *  ----------------   *
40 ' *                     *
50 ' * scrolt het onderste *
60 ' *  deel van screen 2  *
70 ' *     naar boven      *
80 ' *                     *
90 ' **********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDEA0
130 FORI=&HDEA0 TO &HDEC5: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 COLOR 15,1,1:SCREEN 2
150 LINE (50,10)-(200,50),13,B
160 LINE (50,70)-(200,110),13,B
170 LINE (50,130)-(200,170),13,B
180 OPEN"GRP:" AS #1
190 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
200 PRESET(95,90):PRINT#1,"SCROLL"
210 PRESET(88,150):PRINT#1,"ROUTINES"
220 CLOSE#1
230 X=USR2(0):GOTO230
240 DATA 21,00,1A,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
250 DATA ED,B0,F3,21,20,D5,11,00,1A,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ***********************
20 ' *  SCROLL BOTTOM UP    *
30 ' *  ----------------    *
40 ' *                      *
50 ' * scrolt het onderste *
60 ' *  deel van screen 1  *
70 ' *      naar boven      *
80 ' *                      *
90 ' ***********************
100 '
110 CLEAR 200,&HD4FF
120 DEF USR2=&HDEA0
130 FORI=&HDEA0 TO &HDEC5: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
140 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
150 A$="&&&&&&&&&&&&&&&&&&"
160 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
170 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
180 LOCATE4,11:PRINT"&      SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
190 LOCATE4,19:PRINT"&     ROUTINES     &
":LOCATE4,21:PRINTA$
200 X=USR2(0):GOTO200
210 DATA 21,00,1A,11,00,D5,01,00,01,CD,5
9,00,21,00,D5,11,00,D6,01,1F,00
220 DATA ED,B0,F3,21,20,D5,11,00,1A,01,0
0,01,CD,5C,00,FB,C9
```

```
10 ' ******************
20 ' *  SCROLL DOWN    *
30 ' *  -----------    *
40 ' *                 *
50 ' * scrolt screen 2 *
60 ' *  naar beneden   *
70 ' *                 *
80 ' ******************
90 '
100 CLEAR 200,&HA7FF
110 DEF USR0=&HDD50
120 FORI=&HDD50 TO &HDD98: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 COLOR 15,1,1:SCREEN 2
140 LINE (50,10)-(200,50),13,B
150 LINE (50,70)-(200,110),13,B
160 LINE (50,130)-(200,170),13,B
170 OPEN"GRP:" AS #1
180 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
190 PRESET(95,90):PRINT#1,"SCROLL"
200 PRESET(88,150):PRINT#1,"ROUTINES"
210 CLOSE#1
220 X=USR0(0):GOTO220
230 DATA 21,00,00,11,00,A8,01,00,18,CD,5
9,00,21,00,20,11,00,C0,01,00,18
240 DATA CD,59,00,21,00,A8,11,00,01,01,0
0,17,CD,5C,00,21,00,C0,11,00,21
250 DATA 01,00,17,CD,5C,00,21,00,BF,11,0
0,00,01,00,01,CD,5C,00,21,00,D7
260 DATA 11,00,20,01,00,01,CD,5C,00,C9
```

```
10 ' *******************
20 ' *  SCROLL DOWN    *
30 ' *  -----------    *
40 ' *                 *
50 ' * scrolt screen 1 *
60 ' *  naar beneden   *
70 ' *                 *
80 ' *******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR2=&HDD50
120 FORI=&HDD50 TO &HDD74: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
140 A$="&&&&&&&&&&&&&&&&&&&"
150 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
160 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
170 LOCATE4,11:PRINT"&      SCROLL     &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
180 LOCATE4,19:PRINT"&     ROUTINES    &
":LOCATE4,21:PRINTA$
190 X=USR2(0):GOTO190
200 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,E0,D7,11,00,18,01,1F,00
210 DATA CD,5C,00,21,00,D5,11,20,18,01,E
0,02,CD,5C,00,C9
```

```
10 ' *******************
20 ' *   SCROLL UP     *
30 ' *   ----------    *
40 ' *                 *
50 ' * scrolt screen 2 *
60 ' *    naar boven   *
70 ' *                 *
80 ' ******************8
90 '
100 CLEAR 200,&HA7FF
110 DEF USR0=&HDD50
120 FORI=&HDD50 TO &HDD98: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 COLOR 15,1,1:SCREEN 2
140 LINE (50,10)-(200,50),13,B
150 LINE (50,70)-(200,110),13,B
160 LINE (50,130)-(200,170),13,B
170 OPEN"GRP:" AS #1
180 PRESET(70,30):PRINT#1,"TRUUKS & TIPS
"
190 PRESET(95,90):PRINT#1,"SCROLL"
200 PRESET(88,150):PRINT#1,"ROUTINES"
210 CLOSE#1
220 X=USR0(0):GOTO220
230 DATA 21,00,00,11,00,A8,01,00,18,CD,5
9,00,21,00,20,11,00,C0,01,00,18
240 DATA CD,59,00,21,00,A9,11,00,00,01,0
0,17,CD,5C,00,21,00,C1,11,00,20
250 DATA 01,00,17,CD,5C,00,21,00,A8,11,0
0,17,01,00,01,CD,5C,00,21,00,C0
260 DATA 11,00,37,01,00,01,CD,5C,00,C9
```

```
10 ' *******************
20 ' *     SCROLL UP    *
30 ' *     ---------    *
40 ' *                  *
50 ' * scrolt screen 1 *
60 ' *    naar boven    *
70 ' *                  *
80 ' *******************
90 '
100 CLEAR 200,&HD4FF
110 DEF USR1=&HDD00
120 FORI=&HDD00 TO &HDD24: READ A$: POKE
 I,VAL("&H"+A$): NEXT I
130 KEY OFF:COLOR15,1,1:SCREEN1:VPOKE&H2
004,209
140 A$="&&&&&&&&&&&&&&&&"
150 LOCATE4,1:PRINTA$:LOCATE4,3:PRINT"&
 TRUUKS & TIPS  &"
160 LOCATE4,5:PRINTA$:LOCATE4,9:PRINTA$
170 LOCATE4,11:PRINT"&        SCROLL      &
":LOCATE4,13:PRINTA$:LOCATE4,17:PRINTA$
180 LOCATE4,19:PRINT"&      ROUTINES     &
":LOCATE4,21:PRINTA$
190 X=USR1(0):GOTO190
200 DATA 21,00,18,11,00,D5,01,00,03,CD,5
9,00,21,00,D5,11,E0,1A,01,1F,00
210 DATA CD,5C,00,21,20,D5,11,00,18,01,E
0,02,CD,5C,00,C9
```

De grafische cursor

Als de computer zich in een tekst-scherm bevindt, ziet men steeds de cursor op het scherm. Zo heeft men ook de mogelijkheid in een programma deze cursor naar een bepaalde plaats op het scherm te sturen of de X en Y positie op te vragen. Het positioneren van de cursor doet men dan met de instruktie "LOCATE x,y", waarbij x de X-as en y de Y-as voorstelt. Het opvragen van de X positie doet men met de instruktie 'POS (Ø)' en de Y positie doet men met de 'CSRLIN'.
In de grafische schermen is er eveneens een cursor. Deze cursor kan men in tegenstelling tot de tekst-schermen niet zien. Ook de grootte is aanzienlijk minder. Hij is slechts 1 pixel groot. Net als in de tekst-schermen kan ook deze cursor op een bepaalde plaats op het scherm gezet worden. Dit doet men door middel van de instruktie "DRAW"BM x,y", waarbij x voor de X-as en y voor de Y-as staat. Voor MSX-Basic is niet in een instruktie voorzien.

Om de cursorpositie toch te weten te komen kan men de volgende subroutine in een programma inbouwen. Nadat het programma de subroutine verlaten heeft, zal de variabele X1 gelijk zijn aan de X positie van de grafische cursor en zal de variabele Y1 gelijk zijn aan de Y positie van de grafische cursor.

De programmalijst:

```
10 ' **********************
20 ' *  GRAFISCHE CURSOR  *
30 ' *  ----------------  *
40 ' **********************
50 '
```

```
1000 DRAW"U0":T1=PEEK(&HF92A)+256*PEEK(&
HF92B)
1010 Y1=INT(T1/256)*8:A1=T1MOD256:X1=INT
(A1/8)*8
1020 FORI=0TO7:IF PEEK(&HF92C)=2^ITHENX1
=X1+7-I
1030 NEXT:Y1=(A1MOD8)+Y1
1040 RETURN
```

Reaktietest

Het volgende programma dient om uw reaktie-vermogen en ook uw typvaardigheid uit te testen.
Als men het programma opstart zal er een willekeurige tijd niets gebeuren. Na een tijdje verschijnt er op een willekeurige plaats op het scherm een willekeurige letter. Het is de bedoeling zo snel mogelijk deze letter in te toetsen, waarna uw score op het scherm gezet wordt. Het is de bedoeling een minimum aan punten te verzamelen.

De programmalijst:

```
10 ' *****************
20 ' *  REAKTIE TEST  *
30 ' *  ------------  *
40 ' *****************
50 '
60 R=RND(-TIME):POKE&HFCAB,255
70 SCREEN0:T=INT(RND(1)*300):K=INT(RND(1
)*25):P=INT(RND(1)*959)
80 ON INTERVAL=T GOSUB 100:INTERVAL ON
90 GOTO90
100 VPOKEP,K+65:TIME=0
110 I$=INKEY$:IF I$=""THEN110 ELSE IF AS
C(I$)<>K+65THEN110
120 CLS:PRINTTIME;"PUNTEN"
```

Aantekeningen

Aantekeningen

Aantekeningen

Aantekeningen

Aantekeningen

Aantekeningen

In de serie MSX truuks en tips

In de serie MSX truuks en tips worden korte, krachtige oplossingen gepresenteerd voor uw programmeerproblemen. Geen flauwekulletjes maar stukjes programmatuur die u nodig heeft en die u zo kunt toepassen.
Dat in MSX Basic de meeste problemen bijzonder effektief zijn aan te pakken, bewijst de inhoud van dit boek.
Wij zijn ons er echter van bewust dat er bij u programmeerproblemen kunnen leven die in deze serie nog niet werden behandeld.
Wanneer u een of meer van deze problemen heeft en u daarbij vermoedt dat er veel mede-amateurs met dezelfde problemen zitten (en dat tweede is natuurlijk erg belangrijk), zet uw programmeerproblemen dan eens op papier en stuur ze naar:

Uitgeverij Stark-Texel
Postbus 302
1794 ZG Oosterend

en wie weet wordt uw probleem in een volgend deel vakkundig opgelost...
Als uw vraag of probleem in de vorm van een programma-oplossing door ons wordt uitgewerkt, krijgt u het komplete boekje toegestuurd met daarbij nog een aardige verrassing...

Tot ziens in deel 6...

Nederlandstalige MSX handboeken

MSX BASIC handboek voor iedereen, door A.C.J. Groeneveld
Een kompleet nederlandstalig handboek voor iedere MSX computergebruiker. Dit handboek omvat een volledige behandeling van het MSX-basic in het Nederlands. Het handboek geeft een antwoord op elke vraag die een programmeur, van welke scholing ook, over het MSX-basic zou kunnen stellen. De volledige syntaxisbehandeling rekent af met onzekerheden of een bepaalde schrijfwijze nu wel of niet is toegestaan. De duidelijke beschrijving geeft per sleutelwoord aan, welke de funktie hiervan is. De laatste mogelijk nog aanwezig onduidelijkheden worden vervolgens door de opgenomen, zinvolle voorbeelden weggenomen ISBN 90 6398 1007

MSX ZAKBOEKJE door Wessel Akkermans
Een vlot geschreven naslagwerk na of naast het handboek. U vindt er o.a. in: niet computergerichte tabellen; de MSX-BASIC instructieset; diverse tabellen die het BASIC-programmeren kunnen versnellen; de Z80 instructieset; hardware-gegevens (connectoren) en een aantal programmaatjes ISBN 90 6398 888 5

MSX DISK handboek voor iedereen, door A.C.J. Groeneveld
Handboek voor diskdrivebezitters om naast het grote handboek te gebruiken. Een zeer volledige behandeling van het disk-gebeuren zelf en de specifieke disk kommando's, uitgebreid met voorbeelden, tabellen en overzichten. Het handboek is aangevuld met interessante programma's, waaronder een tekentafelprogramma en een basisprogramma voor basisonderhoud ISBN 90 6398 407 3

MSX PRAKTIJKPROGRAMMA'S door Wessel Akkermans
Praktische programma's met waar nodig eerst een stukje theorie. Erg handig bij het maken van uw programma's. Een greep uit de onderwerpen: priemgetallen; zoeken en sorteren; trefwoordenlijsten; converteren van getallen; enz. ISBN 90 6398 437 5

MSX QUICK DISK handboek voor iedereen, door A.C.J. Groeneveld
Hèt handboek voor iedere QUICK DISK gebruiker. Uitvoerige behandeling van de sleutelwoorden aangevuld met duidelijke voorbeelden met listing ISBN 90 6398 254 2

MSX DOS handboek voor iedereen, door A.C.J. Groeneveld
Dit handboek geeft u op een heldere wijze een totaalbeeld van de mogelijkheden van het MSX-DOS. Ook is dit handboek voorzien van een inleiding op het begrip 'operating system' en dus echt een handboek voor iedereen ISBN 90 6398 674 2

MSX LEERBOEKEN
door Wessel Akkermans en Piet den Heijer
De serie MSX leerboeken geeft een komplete kursus MSX-BASIC programmeren, in drie delen. Deze leerboeken zijn gericht op de beginnende programmeur. De moeilijkheidsgraad van de leerstof wordt dan ook slechts geleidelijk hoger. De gebruikte voorbeelden zijn zo praktisch mogelijk gekozen. Hierdoor kunnen al in een vroeg stadium bruikbare programma's worden gemaakt. Dit zal de lezer/leerling er toe aansporen om verder te gaan. Aan het eind van ieder deel is een groot voorbeeldprogramma opgenomen. Dit programma laat zien waartoe de lezer/leerling na bestudering van het betreffende leerboek in staat zal zijn.
Bij ieder leerboek is een afzonderlijk –Opdrachten en uitwerkingen– boekje te verkrijgen. In deze boekjes staan, in volgorde van de hoofdstukken uit het leerboek, vragen en opdrachten met antwoorden en uitwerkingen. Een unieke serie leerboeken voor een ieder die meer over MSX wil weten en het betere werk met zijn computer wil maken.

MSX Basic leerboek deel 1 - ISBN 90 6398 649 1
Opdrachten bij deel 1 - ISBN 90 6398 596 7
MSX Basic leerboek deel 2 - ISBN 90 6398 769 2
Opdrachten bij deel 2 - ISBN 90 6398 556 8
MSX DOS leerboek deel 3 - ISBN 90 6398 519 3
Opdrachten bij deel 3 - ISBN 90 6398 516 9

MSX Verder uitgediept door H. Klopper
Eindelijk een Nederlandstalig boek over het altijd in de mist gehulde onderwerp – PEEKS EN POKES. In dit boek staan alle belangrijke RAM en VRAM adressen. De video chip en zijn registers worden volledig uitgelegd. Maar ook hoe men een machinetaal programma van cassette naar disk kan schrijven. Bovendien een diskloader utility en een uiterst geavanceerde programma beveiliging. Tenslotte zijn er een aantal interessante programma's opgenomen, waaronder een wereldkaart, waarmee verder kan worden geëxperimenteerd. Elke MSX gebruiker kan in dit boek iets van zijn gading vinden en nieuws leren.

ISBN 90 6398 447 2

MSX Machinetaal handboek door H. Klopper en M. Le Belle
Hoewel een MSX computer over een krachtig Basic beschikt, is het toch handig tijdens het programmeren de grondbeginselen van machinetaal te kennen. Daarvoor is dit boek een goede gids. De zaken worden niet puur theoretisch maar ook aan de hand van duidelijke voorbeelden, die direkt bruikbaar zijn, uitvoerig uitgelegd. Enkele onderwerpen zijn verder – scroll routine –machinetaal software (ook in disk Basic) op casette zetten –disassembler –Z80 essembler instrukties –lijst van ROM-routines –alle hook-adressen –bespreking van Basic tokens en een kompleet token-overzicht. Hèt handboek voor iedere MSX programmeur die zijn computer ten volle wil benutten.

ISBN 90 6398 735 8

MSX TRUUKS EN TIPS
door A.C.J. Groeneveld
Hoe laat ik de computer een cirkel arceren, hoe tover ik mijn computer om in een elektronisch orgeltje, hoe maak ik een mooie intro voor een spelletje. Allemaal vraagstukken die zich lastig laten programmeren maar die iedere MSX-er toch graag opgelost wil zien.
Dit boekje staat boordevol truuks en tips, allemaal in gewoon MSX basic geschreven. Bladerend door dit boek komt u tot de ontdekking dat er voornamelijk korte maar uiterst krachtige en bijzonder goed bruikbare routines zijn opgenomen. Dit boekje geeft kort maar krachtig een antwoord op al uw programmeervragen.

deel 1 ISBN 90 6398 900 8
deel 2 ISBN 90 6398 340 9
deel 3 ISBN 90 6398 910 5

SOFTWARE PLUS IN MSX

INTROTAPE MSX door A.C.J. Groeneveld
Heeft u nog maar net een MSX computer gekocht en wilt u graag weten wat de computer kan en hoe u hem kunt leren programmeren? Deze cassette introduceert MSX op een uiterst vriendelijke en onderwijzende manier. U krijgt instructies hoe u de computer aan moet sluiten en de tape laden. Daarna volgt een demonstratie van de mogelijkheden in MSX, zoals het tekenen van sprites en het werken met de driestemmige toongenerator. Het geheel wordt afgesloten met twee 'les' gedeeltes. In anderhalf à drie uur weet u wat de MSX computer is, wat hij kan, en heeft u haast ongemerkt al wat regels geprogrammeerd.

ISBN 90 6398 148 1

MSX SCRIPT door Ton Weijters
Een menugestuurde nederlandstalige tekstverwerker. Het programma is geschikt om efficiënt grotere of kleinere teksten te bewerken. Paginaindeling (regellengte, paginalengte, marge, inspringen, centreren, enz.) wordt door het programma verzorgd. Dit geldt ook voor de paginatelling, toptitel en het eventueel uitvullen van de regels. Ook corrigeren, zoeken, string-substitutie, blokken tekst verplaatsen, copiëren of verwijderen, onderstrepen en vet zetten, is mogelijk met dit programma.

ISBN 90 6398 189 9

MSX DRAWS door A.C.J. Groeneveld
Een tekenprogramma in MSX basic, waarmee u al binnen 10 minuten uw eerste tekening kunt maken. Draws werkt erg vriendelijk en maakt gebruik van alle grafische mogelijkheden van de MSX computer. U kunt met Draws zowel technisch als creatieve tekeningen maken. Het programma heeft een effectief bereik van ruim 30.000 bij 30.000 puntjes met mogelijkheden als lijnen, cirkels, krommen, inkleuren, vergroten, verkleinen, verschuiven, verdraaien en andere tekeningen invoegen

ISBN 90 6398 754 4

MSX2 Basic handboek
A.C.J. Groeneveld, ISBN 90 6398 221 6, 507 pagina's, prijs f 56,50
Wie denkt over de aanschaf van een MSX2 computer, of er al een heeft, kan niet zonder het 507 pagina's tellende MSX2 Basic handboek. Alles over MSX2 Basic, de grafische- en geluidsmogelijkheden en de computer zelf. Met 288 voorbeeldprogramma's.

MSX2 Disk/Dos Uitbreidingshandboek
A.C.J. Groeneveld, ISBN 90 6398 222 4, 172 pagina's, prijs f 37,50
Omvat een volledige behandeling van het MSX2 Disk Basic en het MSX DOS operating system, voorafgegaan door een zeer duidelijke inleiding tot de fenomenen disk en operating system. Verder praktische tabellen, duidelijke afbeeldingen en zinvolle voorbeelden. Een standaardwerk dat naast elke MSX2 Disk computer zou moeten liggen.

MSX2 Utility-Toepassingshandboek
A.C.J. Groeneveld, ISBN 90 6398 223 2, 144 pagina's, prijs f 29,75
Een verzameling programma's die voor elke MSX-er onontbeerlijk zijn. Een aantal van de mogelijkheden met deze programma's: bestandsonderhoud met lijstwerk in iedere vorm op schijf en tape, staaf- en taartdiagrammen, programma's samenstellen met sprites en geluidseffekten, binair manipuleren binnen blokken op schijf. Alle programma's in dit boek zijn geschikt voor zowel MSX als MSX2 computers.

truuks en tips
deel 5

Als MSX-er moet je wel even wat tijd uittrekken om het MSX Basic te leren kennen, maar beheers je het eenmaal, dan kun je met een paar instrukties de meest ongelofelijke dingen doen.
Programmeren is en blijft echter een kunst. Het is niet genoeg om alle woorden van een programmeertaal uit het hoofd te kennen. Nee, de kunst is om vanuit deze bouwstenen een goed funktionerend en foutloos programma te schrijven.
In het op een korte en krachtige wijze oplossen van problemen herkent men de ware programmeur. En ook achter dit vijfde deel zit weer zo'n ware programmeur, die de problemen op de meest doeltreffende wijze weet aan te pakken. Daardoor zijn als het ware allemaal wieltjes ontstaan, die u niet voor de tweede maal hoeft uit te vinden.

Een aantal van de behandelde onderwerpen:

- String-funktie decoder
- Karakter editor
- Tekst op zijn kop
- Goniometrische funkties
- Scroll-routines
- Listbeveiligingen
- Snelle sprites
- Muzikale joystick.

Totaal maar liefst 29 volstrekt verschillende onderwerpen, voor u op een professionele manier volledig uitgewerkt.

MSX truuks en tips. Voor iedere MSX-er een bron van lering en vermaak. Bij ieder deel opnieuw.

ISBN 90 6398 745 5

uitgeverij Stark-Texel